Reflection for PizzaCost Application

Credit Name: CSE2140 2nd Language Programming
Assignment Name: PizzaCost Application

Understanding the Problem

I read the problem carefully and found that there are three costs: **labor**, **rent**, and **materials**. The labor and rent costs are fixed for each pizza, while the materials cost depends on the pizza's **diameter** using the formula:

Materials Cost=0.05×diameter×diameter\text{Materials Cost} = 0.05 \times \text{diameter} \times \text{diameter}Materials Cost=0.05×diameter×diameter

At first, I wasn't sure if labor and rent were per pizza or per inch, but after re-reading, I understood they were **fixed costs per pizza**.

Planning the Solution

Steps I planned:

1. Ask the user for the pizza diameter.
2. Calculate the materials cost.
3. Add labor and rent to the materials cost.
4. Display the total cost.

I used a **Scanner** to get input and **double** variables to handle decimal values.

Implementation

I wrote the program step-by-step:

- Added a Scanner for user input.

- Used constants for labor, rent, and materials rate.
- Calculated and displayed the total cost using `System.out.printf`.

I tested the program with different pizza sizes to check accuracy.

Overcoming Challenges

The hardest part was making sure the formula was correct and formatting the output neatly.
To fix problems, I printed intermediate results to verify calculations.

Learning

I learned how useful **constants** are for fixed values and how to **format output** to make results clear and professional.

Reflection for ObjectHeight Application

Credit Name: CSE2140 2nd Language Programming
Assignment Name: ObjectHeight Application

Understanding the Problem

The task was to calculate the height of an object using:

h=100−4.9×t2h = 100 - 4.9 \times t^2h=100−4.9×t2

where `t` is the time in seconds. The time must be **less than 4.5 seconds**.
At first, I wasn't sure how to enforce this rule, but I decided to use an **if statement** to check the input.

Planning the Solution

Steps I planned:

1. Ask the user for a time value.
2. Check if the time is less than 4.5 seconds.
3. If valid, calculate the height and display it.
4. If invalid, show an error message.

I used a **Scanner** for input and **double** variables for decimals.

Implementation

I started by coding the input, then added the formula.
I tested the program with times like `2.0`, `4.4`, and `4.6` to make sure it worked correctly.

Overcoming Challenges

The hardest part was ensuring the program rejected times **4.5 or greater**.
I solved this by testing often and using print statements to debug.

Learning

I learned how to **validate user input** and how to work with real-world **mathematical formulas** in Java.